SIMPLE PLANT LOCATION PROBLEM

This Hypercard stack was prepared by:
Dennis L. Bricker,
Dept. of Industrial Engineering,
University of Iowa,
Iowa City, Iowa 52242
e-mail: dbricker@icaen.uiowa.edu

author

Given: M candidate locations, N customers

F_i = fixed cost of establishing a plant at site i, i=1,2,...M

SPL

C_{ij} = cost of supplying all demand of customer j from plant i, j=1,2,...N

The Problem: Select a set of plant locations and allocation of customers to plants so as to minimize the total cost.

Note: there are no capacity constraints for a plant which has been selected, and the number of plants is not specified (unlike p-median problem)

ILP models of the SPL problem

Define variables:

$$Y_i = \begin{cases} 1 & \text{if plant site i is selected} \\ 0 & \text{otherwise} \end{cases}$$

$$X_{ij} = \begin{cases} 1 & \text{if plant i serves all demand of customer j} \\ 0 & \text{otherwise} \end{cases}$$

Model #1

$$\textbf{Minimize} \sum_{i=1}^{M} \sum_{j=1}^{N} C_{ij} X_{ij} + \sum_{i=1}^{M} F_i Y_i$$

$$\textbf{s.t.} \sum_{i=1}^{M} X_{ij} = 1 \quad \forall\ j=1,\dots N$$

$$X_{ij} \le Y_i \quad \forall\ i\&j$$

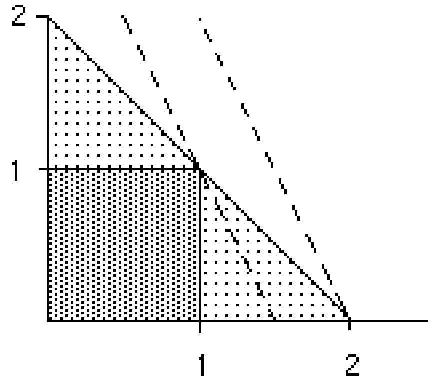
$$Y_i \in \{0,1\},\ X_{ij} \ge 0 \quad \forall\ i\&j$$

Model #2

Replace constraints $X_{ij} \le Y_i \quad \forall\ i\&j$
with aggregated constraints

$$\sum_{j=1}^{N} X_{ij} \le NY_i \quad \forall\ i$$

Models #1 & #2 are equivalent, in that the feasible solution sets are identical....
But-- their LP relaxations (i.e., replacing $\mathbf{Y_i} \in \{\mathbf{0,1}\}$ with $\mathbf{0} \le \mathbf{Y_i} \le \mathbf{1}$) are not!



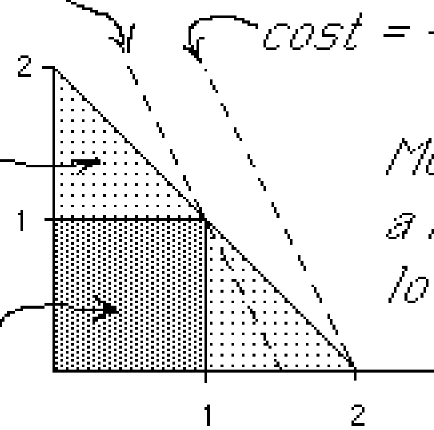
Example

Minimize $-2X_{i1} - X_{i2}$

cost = -3

cost = -4

feasible set for $X_{i1} + X_{i2} \le 2$



feasible set for $X_{i1} \le 1$, $X_{i2} \le 1$

Model #1 provides a higher, "better" lower bound on the optimum!

Model #2 is more "compact", and the LP relaxation is easier to solve.

LP Relaxation of Model #2

At the LP optimum,

$$\sum_{j=1}^{N} X_{ij} \le N Y_i \quad \forall\, i \qquad \text{is "tight"},$$

$$\text{i.e.,} \quad Y_i = \frac{1}{N} \sum_{j=1}^{N} X_{ij}$$

Eliminate Y_i

$$\text{Minimize} \sum_{i=1}^{M} \sum_{j=1}^{N} C_{ij} X_{ij} + \sum_{i=1}^{M} \frac{1}{N} F_i \sum_{j=1}^{N} X_{ij}$$

$$\Longrightarrow \begin{cases} \text{Minimize} \displaystyle\sum_{i=1}^{M} \sum_{j=1}^{N} \left[C_{ij} + \frac{F_i}{N} \right] X_{ij} \\ \text{s.t.} \displaystyle\sum_{i=1}^{M} X_{ij} = 1 \quad \forall\, j=1, \cdots N \\ \qquad X_{ij} \ge 0 \quad \forall\, i \& j \end{cases}$$

The solution is

$$X_{ij}^{*} = \begin{cases} 1 & \text{if } C_{ij} + \dfrac{F_i}{N} \le C_{kj} + \dfrac{F_k}{N} \;\forall i \\ 0 & \text{otherwise} \end{cases}$$

with objective value

$$\sum_{j=1}^{N} \min_{i} \left\{ C_{ij} + \frac{F_i}{N} \right\}$$

Although not a strong bound, this is easily computed:

APL

```
+/ ⌊⌿ C + ⍉(⌽⍴C)⍴F÷N
```

4 = M = # potential plant sites
8 = N = # demand points

Costs

i \ j=	1	2	3	4	5	6	7	8	F
1	4	6	8	9	5	4	3	0	140
2	10	5	10	0	8	10	9	9	120
3	3	5	7	9	4	5	2	3	177
4	8	6	4	7	5	10	8	8	128
D	98	12	7	33	49	33	87	78	

Weak LP Relaxation of Simple Plant Location Problem

The Matrix C + (F÷N)

from \ to	1	2	3	4	5	6	7	8
1	144	146	148	149	145	144	143	140
2	130	125	130	120	128	130	129	129
3	180	182	184	186	181	182	179	180
4	136	134	132	135	133	138	136	136

The LP bound is found by summing the minima in each column

Lower bound provided by weak LP relaxation = 1031.38

Model #3

$$\text{Minimize } \sum_{i=1}^{M} f_i(X_{i1}, X_{i2}, \cdots X_{iN})$$

$$\text{subject to } \sum_{i=1}^{M} X_{ij} = 1 \quad \forall j=1,2,\ldots N$$

$$X_{ij} \geq 0 \;\forall\; i\&j$$

continuous variables only, but objective is discontinuous

where

$$f_i(X_{i1}, X_{i2}, \cdots X_{iN}) = \begin{cases} 0 \text{ if } \sum_{j=1}^{N} X_{ij} = 0 \\ F_i + \sum_{j=1}^{N} C_{ij} X_{ij} \text{ otherwise} \end{cases}$$

Surrogate Constraint

Define a ***surrogate multiplier*** for each constraint: $U_j,\ j=1,\cdots N;\ \sum_j U_j = 1$

Form a linear combination of the constraints

$$\left.\begin{array}{c} U_1 \times \sum_i X_{i1} = U_1 \times 1 \\ \vdots \\ U_N \times \sum_i X_{iN} = U_N \times 1 \end{array}\right\} \Rightarrow \sum_j U_j \sum_i X_{ij} = \sum_j U_j \Rightarrow \sum_j \sum_i U_j X_{ij} = 1$$

This ***surrogate constraint*** is implied by the original set of constraints, but is less restrictive.

Surrogate Relaxation

We replace the original constraints of Model #3 with the single surrogate constraint:

$$\text{Minimize} \sum_{i=1}^{M} f_i(X_{i1}, X_{i2}, \cdots X_{iN})$$

$$\text{subject to} \quad \sum_j \sum_i U_j X_{ij} = 1$$

$$X_{ij} \geq 0 \ \forall \ i\&j$$

Because the objective function is *concave*, the theory of nonlinear programming assures us that an extreme point of the feasible region (i.e., a *basic* solution) is optimal, so only a single variable is ≠ 0.

For example, $X_{ij} = \begin{cases} 1/U_q & \text{if } i=p,\ j=q \\ 0 & \text{otherwise} \end{cases}$

with cost $F_p + C_{pq} \times 1/U_q$

for some p and q.

Therefore, we can solve the surrogate relaxation by enumerating the MxN basic solutions, and selecting the least cost solution:

$$S(U) = \underset{i,j}{\text{minimum}} \left\{ F_i + \frac{C_{ij}}{U_j} \right\}$$

Because the optimal solution of the original SPL problem is feasible in this surrogate relaxation,

$$S(U) \leq \text{optimum of SPL problem}$$

for all $U = (U_1, U_2, \ldots U_N)$

Surrogate Dual Problem

Since for each U, S(U) gives us a lower bound on the SPL optimal value,
select the surrogate multipliers U to give us the "best", i.e., greatest lower bound:

$$\hat{S} = \text{maximum } S(U) \quad \text{s.t. } \sum_j U_j = 1$$

Use of Surrogate Dual bound in a Branch-&-Bound algorithm

Given a value **V** (e.g., the incumbent solution), we can fathom a subproblem if its surrogate dual value $\hat{S}$ exceeds V, and this may be tested without explicitly computing $\hat{S}$:

$$\hat{S} \geq V \Longleftrightarrow \exists\, U=(U_1,\ldots U_N) \text{ such that } \begin{cases} V \leq F_i + C_{ij}/U_j & \forall\, i\&j \\ \sum_j U_j = 1 \end{cases}$$

Assuming $F_i < V$, this is equivalent to

$$\begin{cases} U_j \leq \dfrac{C_{ij}}{V - F_i} \quad \forall i\&j \\ \sum_j U_j = 1 \end{cases}$$

which clearly has a solution if and only if the least upper bounds of U_j, j=1,...N, have a sum ≥ 1:

$$\hat{S} \geq V \Longleftrightarrow \sum_j \min_i \left\{ \frac{C_{ij}}{V - F_i} \right\} \geq 1$$

$$\frac{C_{ij}}{V - F_i}$$

0.44	0.08081	0.06285	0.3333	0.275	0.1481	0.2929	0
1.076	0.06586	0.07684	0	0.4303	0.3622	0.8595	0.7706
0.3443	0.07026	0.05738	0.3478	0.2295	0.1932	0.2037	0.274
0.8682	0.07973	0.03101	0.2558	0.2713	0.3654	0.7708	0.691

Sum: $$\sum_j \min_i \left\{ \frac{C_{ij}}{V - F_i} \right\} = 1.023$$

The conclusion of the comparison test is:

$$\hat{S} \geq V \quad (= 1031)$$

By any of several methods, the equation

$$\sum_j \min_i \left\{ \frac{C_{ij}}{\quad - F_i} \right\} = 1$$

may easily be solved for $\hat{S}$ if the actual value of $\hat{S}$ is necessary.

©Dennis Bricker, U. of Iowa, 1997

Surrogate Dual Algorithm

```
Lower bound= 1074, Upper bound= 1449
Estimated duality gap = 25.89%

Upper bound achieved by Y = 1 1 1 1, i.e.,
       opening plants 1 2 3 4

(Not guaranteed to be optimal!)
```


Surrogate Dual Algorithm

Matrix C÷⍉(⌽⍴C)⍴(SD-F)

0.4198	0.0771	0.05997	0.318	0.2624	0.1414	0.2795	0
1.027	0.0629	0.07339	0	0.411	0.346	0.8209	0.736
0.3278	0.0669	0.05464	0.3312	0.2185	0.184	0.194	0.2609
0.8289	0.07612	0.0296	0.2442	0.259	0.3489	0.7358	0.6597

(Y[i]=1 if any column minimum, i.e., Lambda,
is found in row # i of the matrix above)

Surrogate multipliers

j	1	2	3	4	5	6	7	8
Lambda[j]	0.3278	0.0629	0.0296	0	0.2185	0.1414	0.194	0

Theorem If $\mu_{ij} \geq 0$ and $\sum_{j=1}^{N} \mu_{ij} \leq F_i \; \forall i$

then $\sum_{j=1}^{N} \min_i \{C_{ij} + \mu_{ij}\}$ is a ***lower bound***

for the Simple Plant Location problem

Note: If $\mu_{ij} = \frac{F_i}{N} \; \forall i,j$ *, this is the lower bound provided by the LP relaxation of model #2! By appropriate choice of* μ_{ij} *, it may give us a better lower bound.*

Proof: SPL model #1 may be written

$$\Phi = \text{minimum} \sum_{i,j} C_{ij}X_{ij} + \sum_i \left(F_i - \sum_j \mu_{ij}\right)Y_i + \sum_{i,j} \mu_{ij} Y_i$$

$$\text{s.t.} \sum_i X_{ij} = 1, \; X_{ij} \leq Y_i, \; X_{ij} \geq 0, \; Y_i \in \{0,1\} \; \forall i,j$$

$$\Rightarrow \Phi \geq \sum_{i,j} C_{ij}X_{ij} + \sum_{i,j} \mu_{ij}Y_i \geq \sum_{i,j} C_{ij}X_{ij} + \sum_{i,j} \mu_{ij}X_{ij} = \sum_{i,j} (C_{ij} + \mu_{ij})X_{ij}$$

$$\Rightarrow \quad \text{minimum} \sum_{i,j} (C_{ij} + \mu_{ij})X_{ij}$$

$$\text{s.t.} \sum_i X_{ij} = 1, \; X_{ij} \leq Y_i, \; X_{ij} \geq 0, \; Y_i \in \{0,1\} \; \forall i,j$$

must give us a lower bound for SPL, namely

$$\sum_{j=1}^{N} \min_i \{C_{ij} + \mu_{ij}\}$$

The dual problem is, then, to choose the quantities μ_{ij} so as to obtain the ***greatest lower bound***, i.e.,

$$\text{Maximize} \sum_{j=1}^{N} \min_i \{C_{ij} + \mu_{ij}\}$$
$$\text{s.t.} \quad \sum_j \mu_{ij} \le F_i \ \forall\, i$$
$$\mu_{ij} \ge 0 \ \forall\, i,j$$

$$\text{Maximize} \sum_{j=1}^{N} \min_i \{C_{ij} + \mu_{ij}\}$$
$$\text{s.t.} \quad \sum_j \mu_{ij} \le F_i \ \forall\, i$$
$$\mu_{ij} \ge 0 \ \forall\, i,j$$

The LP equivalent:

$$\text{Maximize} \sum_{j=1}^{N} Z_j$$
$$\text{s.t.} \quad Z_j \le C_{ij} + \mu_{ij} \ \forall\, i,j$$
$$\sum_j \mu_{ij} \le F_i \ \forall\, i$$
$$\mu_{ij} \ge 0 \ \forall\, i,j$$

*The dual of this LP is, in fact, the **LP relaxation** of SPL model #1!*

Bilde-Krarup-Erlenkotter [BKE] Algorithm

This algorithm is a dual ascent algorithm for computing good feasible solutions to the dual of the LP relaxation of Model #1.

At each iteration, exactly one μ_{ij} is adjusted to give an improvement in the lower bound. It terminates when no improvement can be obtained by adjusting a single multiplier.

Bilde-Krarup-Erlenkotter Dual Algorithm

```
Step 1: k←1 & Lambda← 294 60 28 0 196 132 174 0

Step 2a: ϵ= 98 0 0 0 0 0 0 0
Lambda[1]= 392
e= 0 0 98 0, LB= 982

Step 2a: ϵ= 98 0 0 0 0 0 0 0
Lambda[2]= 60
e= 0 0 98 0, LB= 982

Step 2a: ϵ= 98 0 21 0 0 0 0 0
Lambda[3]= 49
e= 0 0 98 21, LB= 1003

Step 2a: ϵ= 98 0 21 120 0 0 0 0
Lambda[4]= 120
e= 0 120 98 21, LB= 1123
```

```
Step 2a: ϵ= 98 0 21 120 49 0 0 0
Lambda[5]= 245
e= 0 120 147 21, LB= 1172

Step 2a: ϵ= 98 0 21 120 49 33 0 0
Lambda[6]= 165
e= 33 120 147 21, LB= 1205

Step 2a: ϵ= 98 0 21 120 49 33 30 0
Lambda[7]= 204
e= 33 120 177 21, LB= 1235

Step 2a: ϵ= 98 0 21 120 49 33 30 107
Lambda[8]= 107
e= 140 120 177 21, LB= 1342

Step 3: do not terminate. Set k← 2

Step 2a: ϵ= 0 0 21 120 49 33 30 107
Lambda[1]= 392
e= 140 120 177 21, LB= 1342
```



```
Step 2a: ϵ= 0 0 21 120 49 33 30 107
Lambda[2]= 60
e= 140 120 177 21, LB= 1342

Step 2a: ϵ= 0 0 0 120 49 33 30 107
Lambda[3]= 49
e= 140 120 177 21, LB= 1342

Step 2a: ϵ= 0 0 0 0 49 33 30 107
Lambda[4]= 120
e= 140 120 177 21, LB= 1342

Step 2a: ϵ= 0 0 0 0 0 33 30 107
Lambda[5]= 245
e= 140 120 177 21, LB= 1342

Step 2a: ϵ= 0 0 0 0 0 0 30 107
Lambda[6]= 165
e= 140 120 177 21, LB= 1342
```

```
Step 2a: ϵ= 0 0 0 0 0 0 0 107
Lambda[7]= 204
e= 140 120 177 21, LB= 1342

Step 2a: ϵ= 0 0 0 0 0 0 0 0
Lambda[8]= 107
e= 140 120 177 21, LB= 1342


Lower bound= 1342, Upper bound= 1342
Duality gap = 0%
No Duality Gap!

Upper bound achieved by Y = 1 1 1 0,
       i.e., opening plants 1 2 3
```

Lagrange multipliers

j	1	2	3	4	5	6	7	8
Lambda[j]	392	60	49	120	245	165	204	107

Summary of Results for Example Problem

		gap
Optimal Solution of SPL =	1342	—
LP Relaxation of Model #1 =	1342	0%
Surrogate Relaxation of Model #3 =	1074	20%
LP Relaxation of Model #2 =	1031.38	23%